RESUMO POO

## Conceitos base

* **Classe** - é uma descrição de um conjunto de objetos que compartilham os mesmos atributos, operações, relacionamentos e semântica;
* **Objeto** - É uma entidade capaz de reter um estado (informação) e que oferece um conjunto de informações (comportamento);
* **Abstração** – Consiste em trabalhar um objeto apenas preocupando-se com as suas propriedades principais;
* **Encapsulamento** – Impede o acesso aos dados por meios diferentes dos métodos propostos;
* **Herança** – Permite criar uma nova classe a partir de uma já existente;
* **Polimorfismo –** Permite a classes derivadas de uma classe base invocar os seus métodos;

## Listas

**O que são?**

Coleção genérica de objetos de um tipo específico que podem ser chamados pelo índice. Oferece métodos de pesquisa, classificação e manipulação.

**Sintaxe:**

* **List**<*Tipo*> *Nome* = **new List**<*Tipo*> { };

**Exemplo:** List<**int**> **intList** = new List<**int**> { };

## Observable Collection

**O que são?**

Coleção dinâmica de objetos que fornece notificações quando são adicionados, removidos ou atualizados dados.

**Sintaxe:**

* *Variável* = **new ObservableCollection**<*Nome*> { };

**Exemplo: pessoa** = new ObservableCollection<**Pessoa**> { };

## Métodos Assíncronos

**O que são?**

Métodos que são executados assincronamente, ou seja, permitem a execução de uma tarefa na thread principal enquanto outra tarefa é executada separadamente.

**Sintaxe:**

* *Método* **async void** *NomeObjeto*() { **await** *NomeTarefa*(); }

**Exemplo: private** async void **startButton\_Click**(**object sender, RoutedEventArgs e**){ await **DoSomethingAsync**(); }

## Delegates

**O que são?**

Apontador de métodos.

**Eventos –** Delegate com o return type void.

**Sintaxe:**

* **Public delegate void** *Método*();

**Exemplo:** public delegate void **OnExcesso\_Handler**(**object sender, EventArgs e**);